UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/040,276	11/09/2001	Daniel S. Gritter	POU901034US1	9648

46369 7590 07/12/2006

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
YIGDALL, MICHAEL J	
ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 07/12/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/040,276	GRITTER, DANIEL S.	
	Examiner	Art Unit	
	Michael J. Yigdall	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 April 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *45-64* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *45-64* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is responsive to Applicant's submission filed on April 7, 2006. Claims 45-64 are pending.

Response to Arguments

2. Applicant's arguments have been fully considered but they are not persuasive.

In response to Applicant's arguments against Olsen and Baker individually (see Applicant's remarks, pages 3-5), one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981), and *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

As set forth in the Office action, Olsen teaches automatically restoring a breakpoint (see, for example, column 13, lines 8-12) to a selected step of a modified program (see, for example, column 5, lines 28-30). The breakpoint was initially set to a selected step in the source code of the program (see, for example, column 12, lines 1-3), and the program was subsequently modified such that the selected step is now at a different location within the modified program (see, for example, column 2, lines 63-67). Olsen further teaches that restoring the breakpoint comprises comparing one or more attributes of one or more machine instructions generated for the source code with one or more attributes of an instruction profile created based on the source code (see, for example, column 12, lines 15-63) to determine the different location of the selected step (see, for example, column 12, line 64 to column 13, line 4).

The source code that Olsen discloses is considered a "first version" of source code. What Olsen does not disclose is that the modified program has a "second version" of source code. Accordingly, Olsen does not expressly disclose that in the above comparison step, the one or more machine instructions are "generated for the second version of source code."

Baker, however, teaches a first version of source code that is modified to provide a modified program having a second version of source code (see, for example, column 1, line 64 to column 2, line 8). Moreover, Baker teaches comparing one or more attributes of one or more binary code instructions generated for the different versions of the source code, so as to find similarities in the programs even when the source code is not accessible (see, for example, column 3, lines 17-31). The binary code instructions are disassembled and preprocessed to create an instruction profile that comprises one or more attributes of the instructions (see, for example, blocks 120 and 130 in FIG. 1).

Again, Olsen teaches automatically restoring a breakpoint to a selected step of a modified program. Baker likewise teaches a modified program. In Olsen, an optimization provides the modified program (see, for example, column 2, lines 63-67). In Baker, a modification of the source code provides the modified program (see, for example, column 1, line 64 to column 2, line 8). One of ordinary skill in the art would have been motivated to apply Olsen's teachings regardless of how or why the program is modified. In other words, one of ordinary skill in the art would have been motivated to automatically restore a breakpoint to a selected step of a modified program, regardless of whether it is an optimization (as in Olsen), a modification of the source code (as in Baker), or some other operation that provides the modified program.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baker's instruction profile into Olsen's instruction profile, both as noted above, and to not only compare the one or more attributes of the one or more machine instructions with the one or more attributes of the instruction profile created based on the first version of source code, as Olsen teaches, but to also do so when the machine instructions are generated for a second version of the source code, as Baker suggests. Baker's instruction profile enables such a comparison even when the different versions of source code are not accessible, thus providing more opportunities to apply Olsen's teachings.

Applicant contends that Olsen's program is not absent embedded debug commands (remarks, page 5, last paragraph).

However, Applicant mischaracterizes Olsen's debug information. There is nothing to suggest that the debug information is in any way a "command" that is "embedded" in the program. A compiler emits the debug information; the debug information is not a command that a user somehow inserts into the source code (see, for example, column 8, lines 30-32). Applicant even acknowledges that Olsen shows source code that is without embedded debug commands (remarks, page 5, last paragraph).

Nonetheless, upon a closer inspection of Applicant's specification, the examiner cannot discern any description of "embedded debug commands" or the intent of an "absence" of such commands. Accordingly, a rejection of the claims under 35 U.S.C. 112, first paragraph, is now set forth below.

Applicant contends that there is no teaching or suggestion in the references themselves to make the combination or modification suggested in the Office action (remarks, page 6, second full paragraph), and alleges generally that any justification for the combination provided in the Office action does not indicate where the references expressly teach the combination (remarks, page 6, third full paragraph).

First, however, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988), and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992).

As noted above, Baker expressly teaches modifying a first version of source code to provide a modified program having a second version of source code. Olsen expressly teaches automatically restoring a breakpoint to a selected step of a modified program. Thus, it would have been obvious to one of ordinary skill in the art at the time the invention was made to automatically restore a breakpoint to a selected step of a modified program having a second version of source code.

Applicant alleges generally that the combination or suggested modification appears to be a hindsight reconstruction of Applicant's invention (remarks, page 6, third full paragraph).

However, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention

was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

Applicant contends that Olsen and Baker are each trying to solve different problems, and are not reasonably pertinent to the particular problem with which the inventor is involved (remarks, page 6, last paragraph).

However, the examiner disagrees. Applicant's invention "relates, in general, to the debugging of computer programs, and, in particular, to the restoring of debugging breakpoints, subsequent to modifying code of a program" (specification, page 1, paragraph 0001). Olsen is certainly pertinent to this problem. Olsen is directed to restoring debugging breakpoints subsequent to an optimization modifying the code of a program (see, for example, the abstract). Moreover, part of Applicant's invention involves comparing lines of code and finding matches in the code (drawings, pages 5-6, figures 5A and 5B). Baker is certainly pertinent to this problem. Baker is directed to finding similarities in programs (see, for example, the abstract).

Applicant contends that there is no description, teaching or suggestion of using the source line number of a first version of source code and a length of the first version of source code to determine a starting point from which one or more instructions generated for the second version of source code are to be used in the comparing in order to automatically restore the breakpoint to the selected step within the modified program (remarks, page 7, third full paragraph).

However, as set forth in the Office action, Olsen teaches using compiler information to determine a starting point for the comparison (see, for example, column 12, lines 15-27). The compiler information includes source line numbers and lengths of the source code (see, for

example, column 8, line 30 to column 9, line 14). Again, the source code that Olsen discloses is considered a "first version" of source code, and in view of Baker's teachings, the one or more instructions are generated for a "second version" of source code.

Applicant's argument that the stepwise procedure recited in claims 51, 58 and 64 is not described, taught or suggested in Olsen or Baker (remarks, page 8, first full paragraph) amounts to a general allegation of patentability without specifically pointing out how the language of the claims distinguishes over the references, and therefore does not comply with 37 CFR 1.111(b). Accordingly, this argument is not persuasive.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

4. Claims 45-64 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Specifically, independent claims 45, 53 and 59 recite the limitation, "said program being absent embedded debug commands." Applicant's specification as originally filed does not include any description of "embedded debug commands," and accordingly cannot provide the necessary support for this limitation. Thus, all remaining dependent claims, 46-52, 54-58 and 60-64, are also rejected as being dependent on rejected base claims.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 45-64 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,263,489 to Olsen et al. (art of record, "Olsen") in view of U.S. Patent No. 6,282,698 to Baker et al. (art of record, "Baker").

With respect to claim 45 (previously presented), Olsen discloses a method of restoring debugging breakpoints (see, for example, the abstract), said method comprising:

(a) having a breakpoint that is set to a selected step of a first version of source code of a program (see, for example, column 12, lines 1-3, which shows having a breakpoint that is set to a selected step in the source code of a program), said program being absent embedded debug commands (see, for example, column 4, lines 12-28, which shows illustrative source code that is absent embedded debug commands);

(b) creating an instruction profile for the selected step, said instruction profile comprising one or more attributes of one or more machine instructions generated for the selected step and one or more attributes of zero or more other machine instructions generated for the first version of source code (see, for example, column 12, lines 3-15, which shows calculating an instruction

profile for the selected step that comprises one or more attributes of one or more machine code instructions generated for the selected step and other instructions in the source code); and

(c) automatically restoring the breakpoint to the selected step of a modified program (see, for example, column 13, lines 8-12, which shows automatically restoring the breakpoint, and column 5, lines 28-30, which shows that the breakpoint is restored to the selected step), in response to modification of the first version of source code to provide the modified program, wherein the selected step is at a different location within the modified program (see, for example, column 2, lines 63-67, which shows that the source code was optimized to provide a modified program, and that the selected step does not correspond to the same location).

Olsen discloses that the automatically restoring comprises comparing one or more attributes of the one or more machine code instructions with the one or more attributes of the instruction profile created based on the source code (see, for example, column 12, lines 15-63) to determine the different location (see, for example, column 12, line 64 to column 13, line 4), but does not expressly disclose that the modified program has a second version of source code, and that the automatically restoring comprises comparing one or more attributes of one or more machine instructions generated for the second version of source code with one or more attributes of the instruction profile created based on the first version of source code.

However, Baker discloses comparing one or more attributes of the one or more binary code instructions generated for different versions of source code to find similarities in the programs, even if the source code is not accessible (see, for example, column 3, lines 17-31). In Baker, the first version of source code is modified to provide a modified program having a second version of source code (see, for example, column 1, line 64 to column 2, line 8). Baker

further discloses disassembling the binary code instructions and preprocessing the disassembled instructions to create an instruction profile that comprises one or more attributes of the binary code instructions (see, for example, blocks 120 and 130 in FIG. 1).

One of ordinary skill in the art would have been motivated to apply Olsen's teachings regardless of how or why the program is modified. In other words, one of ordinary skill in the art would have been motivated to automatically restore a breakpoint to a selected step of a modified program, regardless of whether it is an optimization (see, for example, Olsen, column 2, lines 63-67), a modification of the source code (see, for example, Baker, column 1, line 64 to column 2, line 8), or some other operation that provides the modified program.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Baker's instruction profile into Olsen's instruction profile, and to not only compare the one or more attributes of the one or more machine code instructions with the one or more attributes of the instruction profile created based on the first version of source code, as Olsen teaches, but to also do so when the machine code instructions are generated for a second version of source code, as Baker suggests. Baker's instruction profile enables such a comparison even when the different versions of source code are not accessible, thus providing more opportunities to apply Olsen's teachings.

With respect to claim 46 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that the comparing comprises comparing one or more operation codes of the one or more machine instructions generated for the second version of source code with one or more operation codes of the instruction profile to determine which machine instruction of the modified program corresponds most closely to the selected step (see,

for example, Baker, column 7, lines 11-30, which shows that the opcodes of the assembly instructions are included in the instruction profile and are compared).

With respect to claim 47 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that the instruction profile further comprises a source line number for the selected step and a length of the first version of source code (see, for example, Olsen, column 8, line 30 to column 9, line 14, which shows that the instruction profile further comprises compiler information that includes source line numbers and lengths of the source code), and wherein the automatically restoring comprises using the source line number and length to determine a starting point within the modified program to select the one or more machine instructions generated for the second version to be used in the comparing (see, for example, Olsen, column 12, lines 15-27, which shows that a starting point for the comparing is determined from the compiler information).

With respect to claim 48 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that the comparing yields one or more difference counts and a difference count having a smallest value indicates the different location (see, for example, Olsen, column 10, lines 12-16, which shows that the comparing comprises finding the difference in counter values, and column 9, lines 51-54, which shows that the earliest or smallest difference indicates the different location).

With respect to claim 49 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that the different location is identified by a substantial match between one or more attributes of the instruction profile and one or more

attributes of one or more machine instructions of the modified program (see, for example, Olsen, column 10, lines 53-63, which shows that a substantial match between the one or more attributes of the instruction profile and the one or more attributes of the machine code instructions of the modified program identifies the different location).

With respect to claim 50 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that the creating comprises choosing a number of machine instructions to be included in the instruction profile (see, for example, Olsen, column 12, lines 3-14, which shows choosing a number of machine code instructions to include).

With respect to claim 51 (previously presented), the rejection of claim 50 is incorporated, and Olsen in view of Baker further discloses that the choosing comprises:

(a) selecting a number of instructions to be included in a calibration profile (see, for example, Olsen, column 5, lines 8-23, which shows selecting a number of instructions to include in a calibration profile);

(b) generating the calibration profile for a chosen line of the program, said calibration profile having the selected number of instructions for said chosen line (see, for example, Olsen, column 4, line 66 to column 5, line 8, which shows calculating or generating the calibration profile for a chosen location);

(c) comparing one or more attributes of said calibration profile to one or more attributes of at least one line of code of the program to obtain a result (see, for example, Olsen, column 12, lines 15-63, which shows comparing one or more attributes of the calibration profile to one or more attributes of at least one line of code);

(d) determining whether the result is an unambiguous result (see, for example, Olsen, column 5, lines 38-45, which shows an ambiguous result, and column 5, lines 46-51, which shows determining whether the result is unambiguous); and

(e) repeating, zero or more times, said selecting, said generating, said comparing, and said determining until the determining indicates an unambiguous result, wherein the selected number of instructions increases at each iteration (see, for example, Olsen, column 9, lines 19-49, which shows repeating the steps zero or more times until the result is unambiguous), and wherein the selected number of instructions indicates, when there is an indication of an unambiguous result, the number of machine instructions to be included in the instruction profile (see, for example, Olsen, column 10, lines 4-11, which shows indicating the instructions to include in the instruction profile when the result is unambiguous).

With respect to claim 52 (previously presented), the rejection of claim 45 is incorporated, and Olsen in view of Baker further discloses that said automatically restoring is performed by a debugger (see, for example, Olsen, column 3, lines 57-67, which shows the debugger).

With respect to claims 53-58 (previously presented), the claims recite a system that corresponds to the method of claims 45-52 (see the rejection of claims 45-52 above).

With respect to claims 59-64 (previously presented), the claims recite an article of manufacture that corresponds to the method of claims 45-64 (see the rejection of claims 45-52 above).

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (571) 272-3707. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

MY

Michael J. Yigdall
Examiner
Art Unit 2192

mjy

TUAN DAM
SUPERVISORY PATENT EXAMINER